السلام عليكم ورحمة الله ,,,

في مثالنا اليوم سنتعرف على كيفية إستخراج الأوامر من الملف التنفيذي , وسنتعرف على طريقة إستخدام الدوال بداخل البرامج ,,,

مثال : هل شاهدت برنامج تنفيذي يحتوي على زر أمر وعند الضغط علية فإنة يقوم بعمل ما ؟!
مثلا فتح السيدي ,, إظهار نافذة معينة , إغلاق الجهاز , إظهار معلومة معينة ,, والكثير..

هل تريد معرفة الكود أو فكرة الكود المكتوب بداخل هذا الأمر ؟!
هل تريد إحتراف النبش في أكواد الملف التنفيذي ؟!

في الملف المرفق ستجد ملف تنفيذي , وهو الملف الذي سنطبق علية الدرس
وستجد أيضاً ملف مضغوط آخر بالإسم Scode هذا الملف يحتوي على الشفرة المصدرية للملف التنفيذي لكي نقارن بين ماحصلنا علية بالبرمجة العكسية , وبين الكود الأصلي

نبدأ في الموضوع :

شغل الملف التنفيذي , وجرب الأوامر الموجودة في النافذة , وبعد ذلك أغلق البرنامج
ثم قم بإخراج الملف التنفيذي من الملف المضغوط إلى أي مجلد ,

شغل برنامج olly ومن قائمة file ثم Open , تجول في الجهاز وإختر الملف التنفيذي
وبعد أن ينتهي olly من فك التجميع شغل البرنامج المراقب عن طريق المفتاح F9

والآن سنقوم بطريقة جديدة في إستخراج الأوامر , لم تكن موضحة في الأمثلة السابقة ,
بعد أن ترى نافذة البرنامج المراقب قد ظهرت , إرجع إلى olly وإضغط على حرف W
أو من قائمة View ثم Windows

ستظهر لك نافذة جديدة تحتوي على أسماء الأوامر والنوافذ الموجودة في البرنامج , بهذا الشكل

والآن إختر النافذة الأولى وهي النافذة الرئيسية , ثم إضغط على الزر الأيمن للماوس
ستظهر لك قائمة إختر الأمر الخامس Message breakpoint on Classproc

ستظهر لك نافذة لإختيار نقطة توقف على رسائل النظام , بهذا الشكل

بالتأكيد فإننا سنقوم بإختيار الرسالة WM _COMMAND ورقمها 111
وتعني نقطة توقف على أي زر أمر في البرنامج , وتستطيع أن تراقب أي رسالة في النظام

المهم بعد تحديد الرسالة , إضغط على زر OK , لتعود إلى قائمة النوافذ
ستلاحظ تغير لون عنوان النافذة الأولى إلى اللون الوردي , بهذا الشكل

والآن إرجع إلى نافذة البرنامج المراقب , ثم إضغط على أول أمر CMD 01
هذا الأمر يقوم بعرض حافظة الشاشة الموقتة

وعند الضغط علية ستلاحظ توقف التنفيذ عند العنوان address 004010E0
هذا العنوان يمثل عنوان بداية دالة معالجة الرسائل للنافذة الرئيسية ,

عرفنا بداية ونهاية الكود للأمر CMD 1 من خلال switch و case وهذة أوامر مقارنة يعرفها الميرمجين , وإذا كنت لا تعرفها , الحل بسيط عن طريق مفتاح F8 بمجرد إستمرار الضغط ستجد أن البرنامج يقارن الأوامر ثم ينفذ CMD 1 وينتقل التنفيذ

بعد أن حددنا كود الأمر الأول بقي أن نحدد طريقة إستخدام الدوال ,, والطريقة أسهل

في البداية نجد بأنة في كود الأمر CMD 1 إستخدم دالتين , كما هو موضح في الصورة
والدالتين هما : SendMessage و GetActiveWindow
وتلاحظ بأن برنامج olly يبين لك من خلال الخطوط بأن دالة مستخدمة داخل دالة أخرى
ويبين لك بارمترات الدالة SendMessage , بهذا التخطيط
ملاحظة الدالة sendmessage لها أربع بارمترات , وفي لغة الإسمبلي تكتب بالعكس
لاحظ الكود الموضح في الصورة :
كود

```
00040112 PUSH: B          البارمتر الرابع // 0= NULL = or
0F1400040112 PUSH: D      البارمتر الثالث // =0F1400x
.............
1120040113 PUSH: B        الثاني //=112 WM_SYSCOMMAND = or
00401140 :CALL GetActiveWindow
00401146 :PUSH EAX        البارمتر الأول // hWnd //
00401147 :CALL  SendMessageA
```

لو فتحت الكود المصدري للبرنامج لوجدت أن كود الأمر بلغة السي هو
كود

```
SendMessage(GetActiveWindow(),WM_SYSCOMMAND,SC_SCREENSAVE,NULL);
```

وإذا كتبت الكود الموضح بالأرقام كما هو موضح في olly فإن الأمر لن يختلف
لأن المترجم في النهاية يكتب الأرقام , مثال نفس الكود يمكن أن يكتب بلغة السي :
كود

```
SendMessage(GetActiveWindow(),0x112,0xF140,0x0);
```

ويمكن أن يكتب نفس الكود حتى في برامج الفيجول بيسك , مع ملاحظة تغيير الرمز للأرقام الهكس , مثلا الرقم 0x1120 يكتب في الفيجول بيسك h112

وتستطيع إيجاد البارمترات مرتبة وجاهزة عن طريق المكدس ؟!
إستمر في الضغط على F8 إلى أن تصل إلى العنوان 00401147 address
وهو أمر الإستدعاء للدالة sendmessageA وبعد أن يصل التنفيذ إلى هذا العنوان

فقط إطلع على قسم المكدس , وهو القسم الموضح في النافذة اليمنى في الأسفل
لتجد البارمترات مرتبة كما كتبت في الكود المصدري

وبهذا نكون قد إستخرجنا الكود الذي كتب تحت زر الأمر دون رؤية الشفرة المصدرية
وهذة الطريقة تعتبر من أهم الفوائد في البرمجة العكسية (معرفة أكواد البرامج)

لو قمت بإكمال إستخراج الأكواد لبقية الأوامر , سيصادفك شكل آخر من الدوال وهي الدوال الخاصة في البرنامج .. تنقل التنفيذ ليتم تفيذ مجموعة من الدوال وسنأخذ مثال ,,,

الأمر الثاني CMD 02 مشابهة للأول مع إختلاف البارمتر الثالث
ويستخدم لعرض قائمة إبدأ start

الأمر الثالث CMD 03 :
هذا الأمر يشغل نافذة إعدادات العرض عن طريق لوحة التحكم ,, كيف؟
قد تكون هذة الدالة طويلة ,, لأنها تتصل بدالة داخل البرنامج تقوم بدورها بالإتصال بعدة دوال وهذا هو الشكل الثاني للأوامر ,, ولكن الشغلة بسيط !

بما أننا في بداية الموضوع حددنا نقطة توقف على كل الأوامر , فلا نحتاج لأمر آخر
شغل البرنامج المراقب عن طريق F9 , بعد ذلك نفذ الأمر CMD 03
ليتوقف التنفيذ عند بداية معالجة الرسائل ,, اللون الوردي

بعد ذلك إستمر في التنفيذ F8 لتتجاوز دوال المقارنة للرسائل , وسينقلك التنفيذ
إلى بداية كود الأمر CMD 03 عند العنوان address 00401152
وهو يمثل دالة خاصة في البرنامج وتستخدم بارمتر واحد
كود

```
00401152  :PUSH  ASCII "Desk.cpl";
00401157  :CALL    004014F0
```

هذة الأوامر تمثل في الشفرة المصدرية الإنتقال للدالة الداخلية الخاصة في البرنامج
وهي LaunchControlPanelApplet

المهم علّم على العنوان 00401157 وهو يمثل التعليمة CALL
ثم إضغط الزر الأيمن للماوس وإختر الأمر Follow لينقلك البرنامج إلى عنوان جديد
وهو العنوان address 004014F0 , ضع نقطة توقف على العنوان
بإستخدام F2 , ثم نفذ البرنامج بإستخدام F9 , ليتوقف التنفيذ في بداية الدالة

ستلاحظ بأن الدالة تبدأ بدوال خاصة بترتيب النصوص , ما يهمنا هي دالة التشغيل , إستمر في تنفيذ الكود إلى أن تصل إلى الدالة CreateProcess

لترى كيف كتبت الدالة , والبارمترات في مسجل المكدس

أعتقد بأن بارمترات الدالة واضحة , وبنفس الترتيب الموجود في الكود المصدري

بقي آخر بارمترين في الدالة وهما , pStartupInfo و pProcessInfo
هذة البارمترات عبارة عن إتحاد لمجموعة من المتغيرات
ولعرض قيم المتغيرات المستخدمة في الدالة ,,
إختر البنية أو الإتحاد , ثم إضغط مفتاح Enter
أو من خلال الضغط للزر الأيمن للماوس وإختيار Follow in stack

وبهذا نكون قد أنهينا الثلاث أوامر ,, وتعلمنا شيء مهم في البرمجة العكسية

إستخراج أفكار ودوال البرامج

هذة الشغلة لو إستمريت عليها وحاولت نبش أكواد أي برنامج تراة ,,, بعد فترة ستصبح خبير في بنية البرامج وطريقة عمل الأنظمة ,, وأي شيء يحدث أمامك على الشاشة صدقني ستعرف مصدرة ,, وكيف تم عملة دون الحاجة لفك التجميع ؟!!